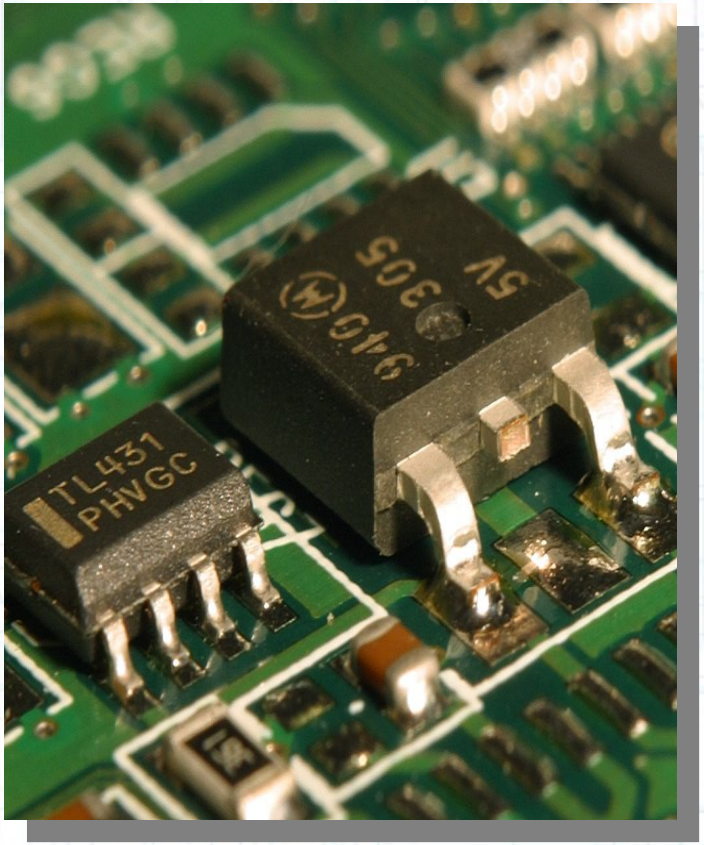
ECE6332 VLSI Design Class Project

Need for Speed:
Process Agnostic Library Migration Automation

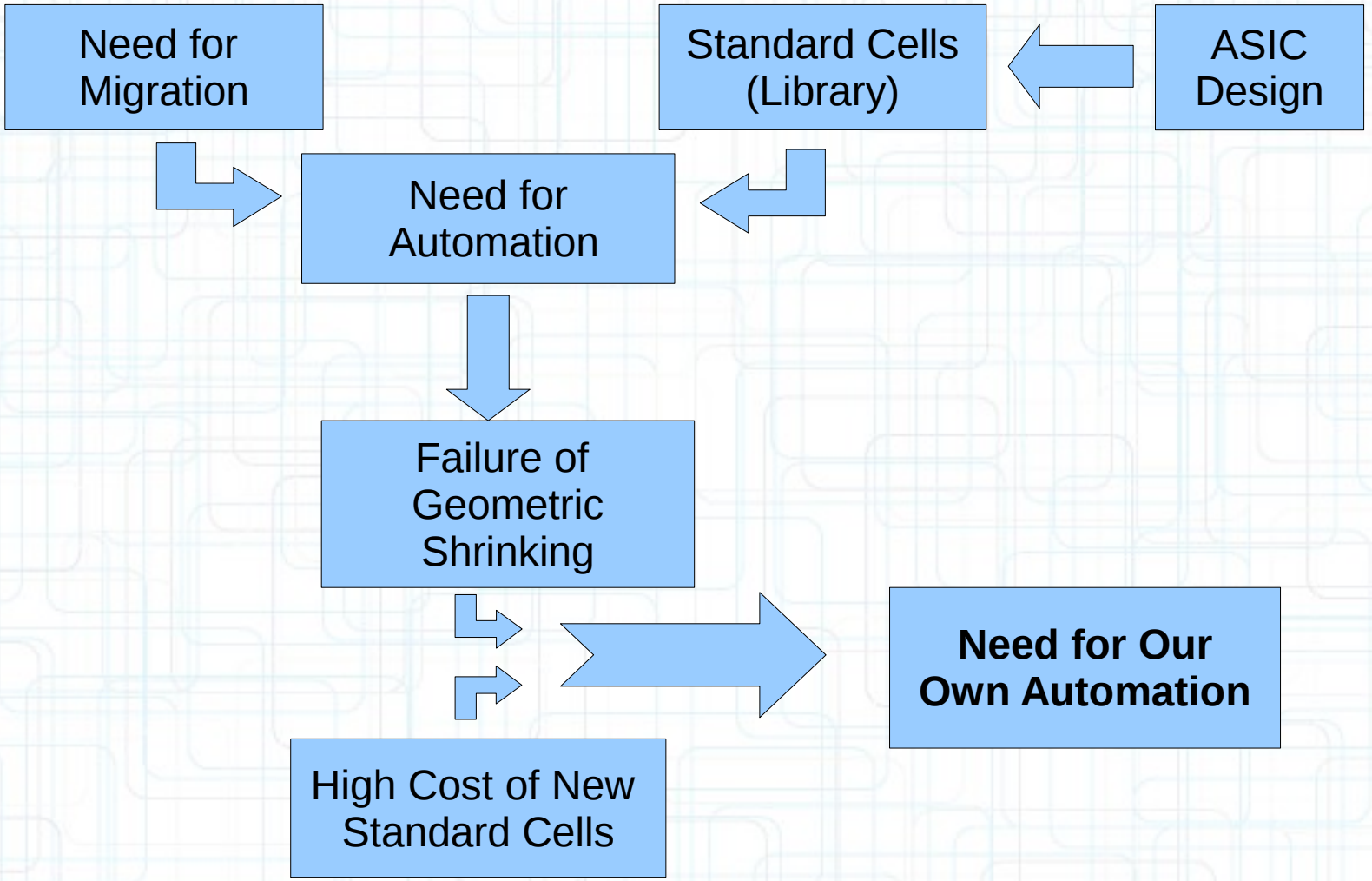
Joseph Murray
Lijun Li

Outline



- Motivation
- Approach
  - PyCell Studio
  - Cadence SKILL
- Comparison
- Summary

Why Process Agnostic Migration Automation



Project Purpose:

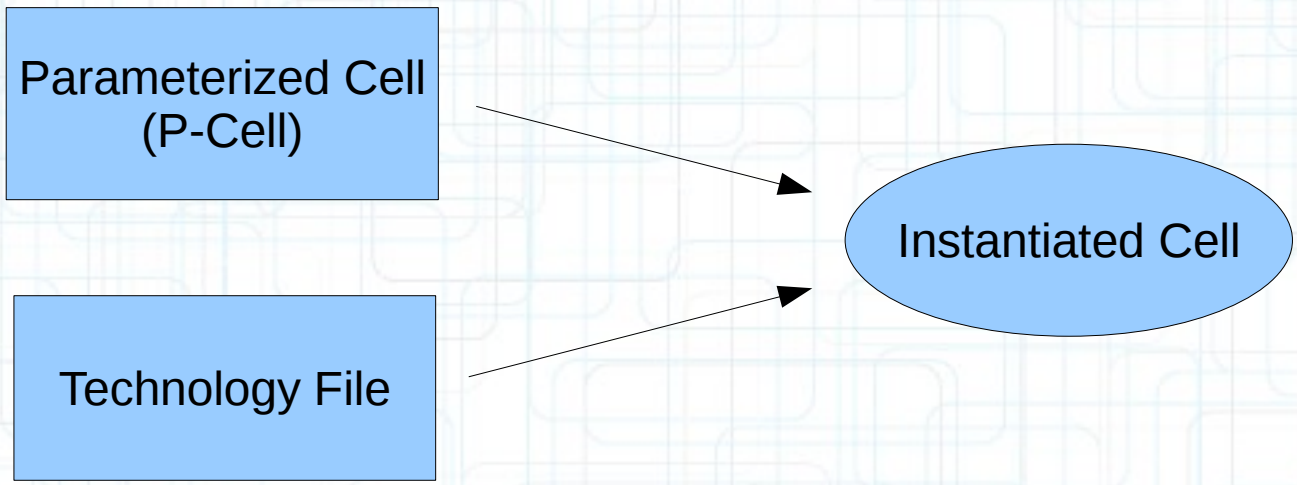
1. Demonstrate the concept
2. Evaluation

PyCell Studio: Design Overview

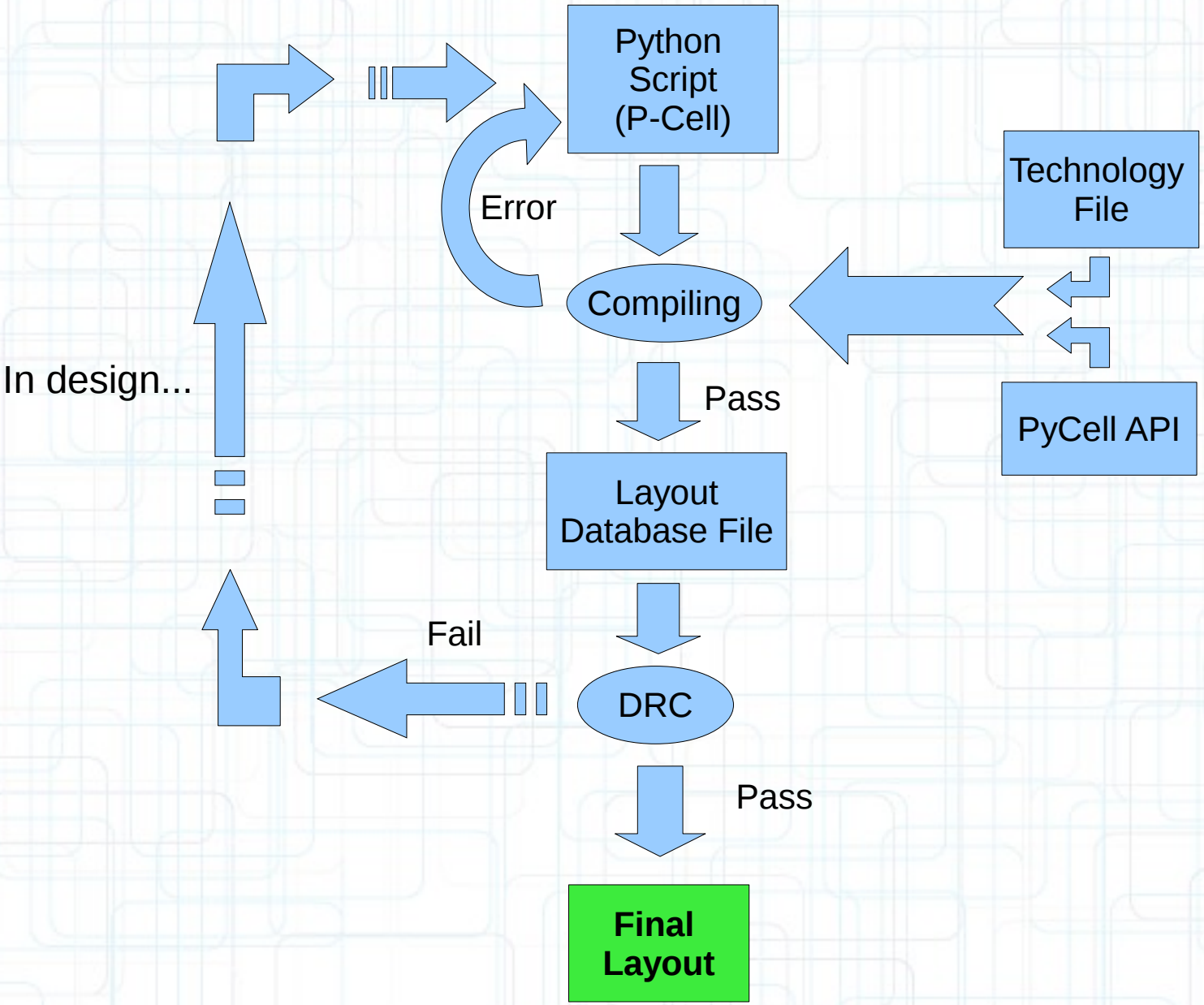
PyCell:

- Based on Python
- Free to use and multiplatform
- Open Access database
- Geared towards parameterized cell design
- **Our home-made PyCell recipe @ ECE Wiki**

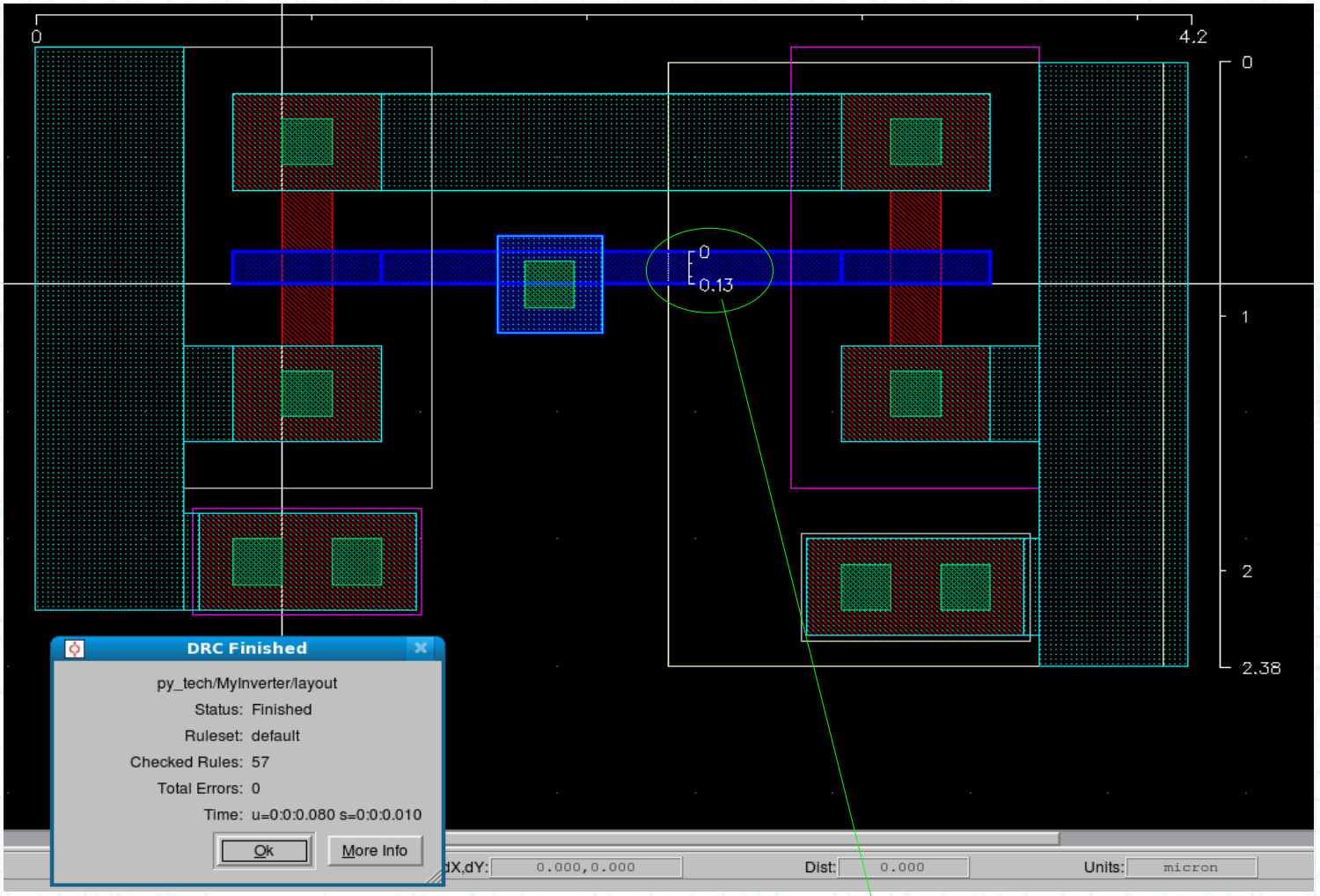
Parameterized Cell Methodology:

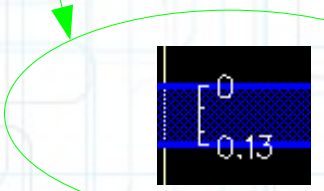


PyCell Studio: Methodology

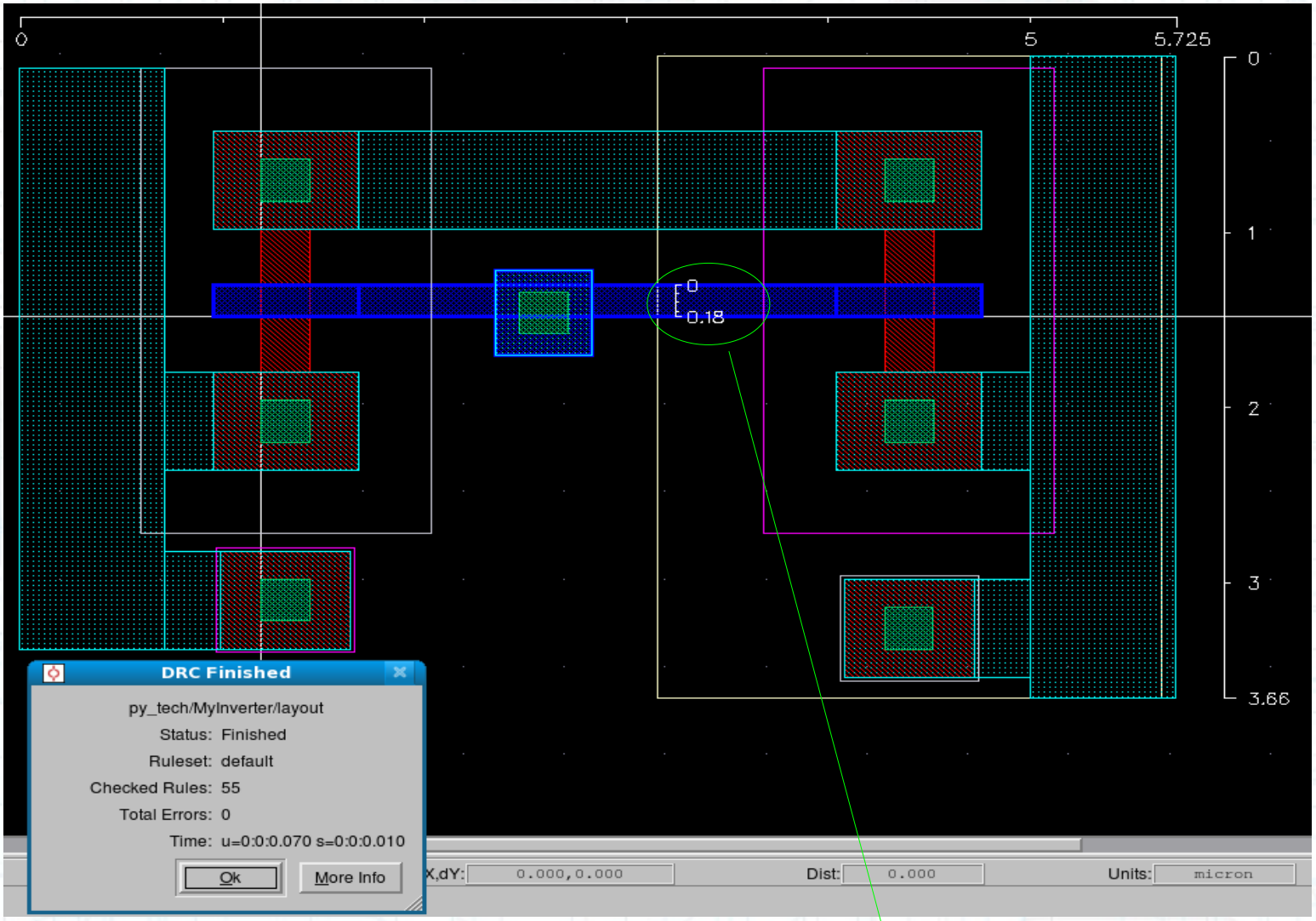


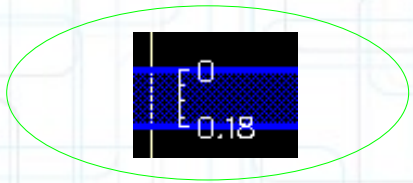
PyCell Studio: Inverter @130nm



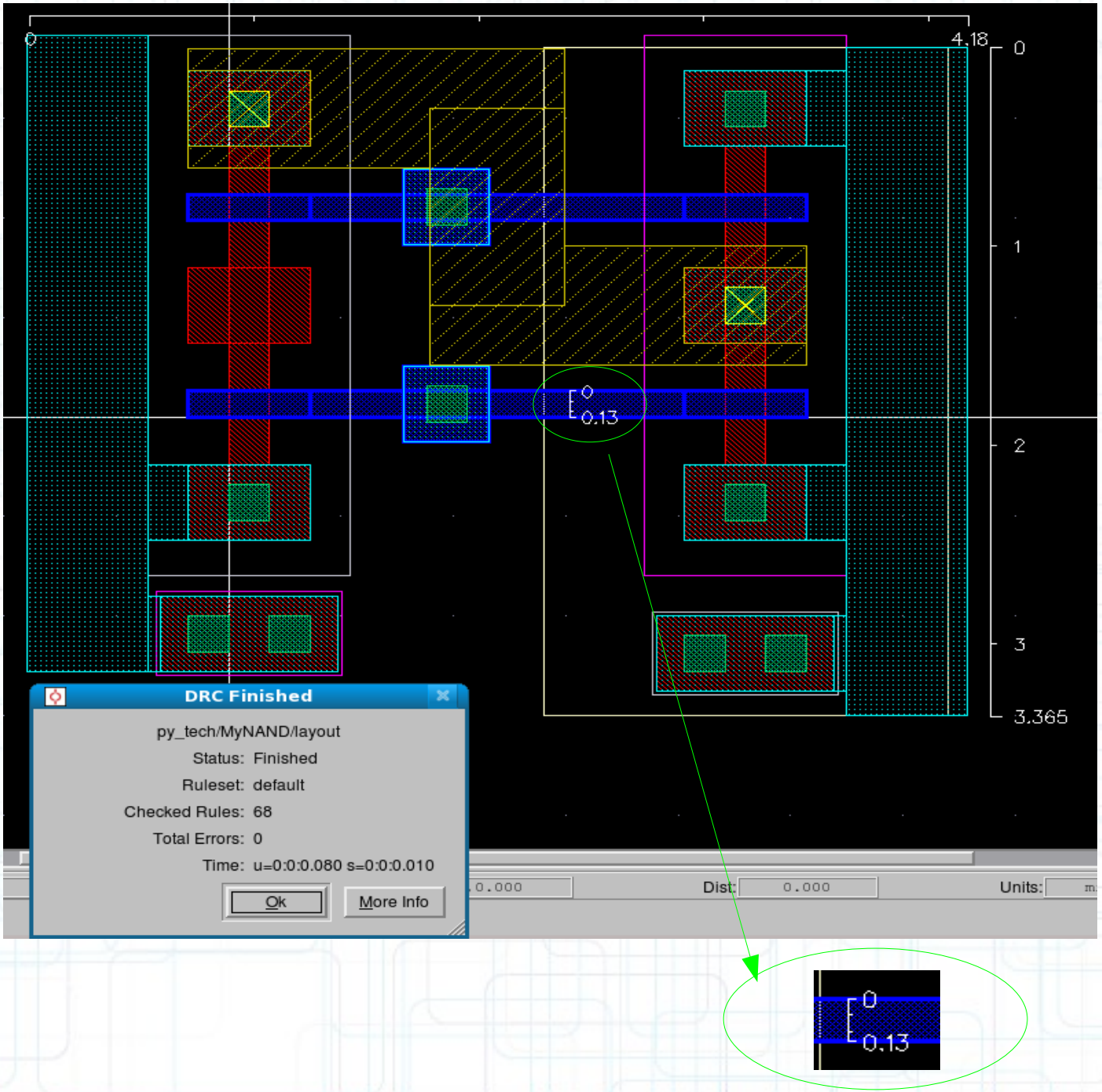


PyCell Studio: Inverter @180nm

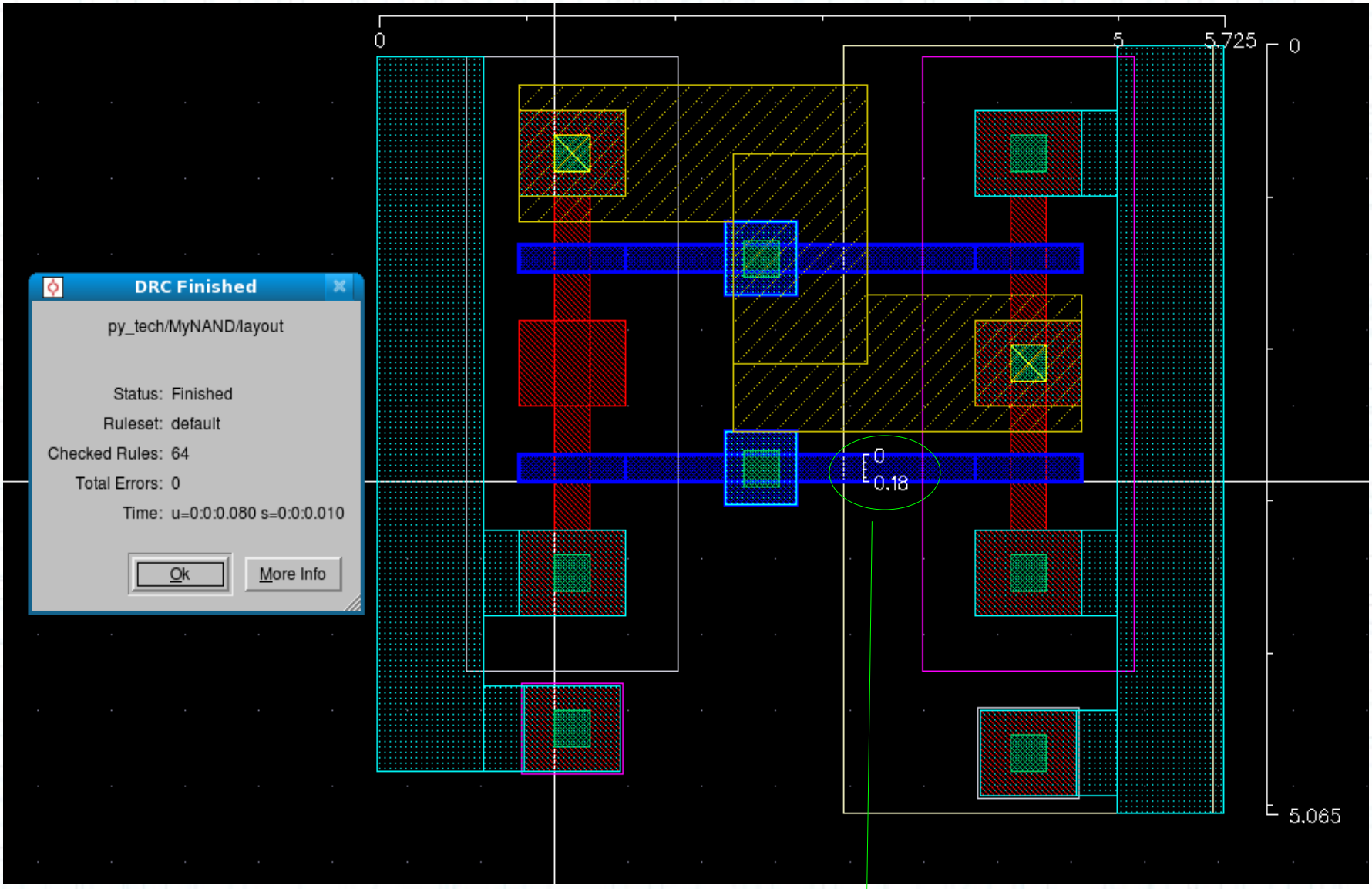


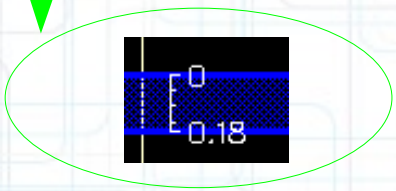


PyCell Studio: NAND @130nm



PyCell Studio: NAND @180nm





PyCell Studio: Pros & Cons

Pros

- DRC automatically met
- Relatively simple design (almost no more than geometry)

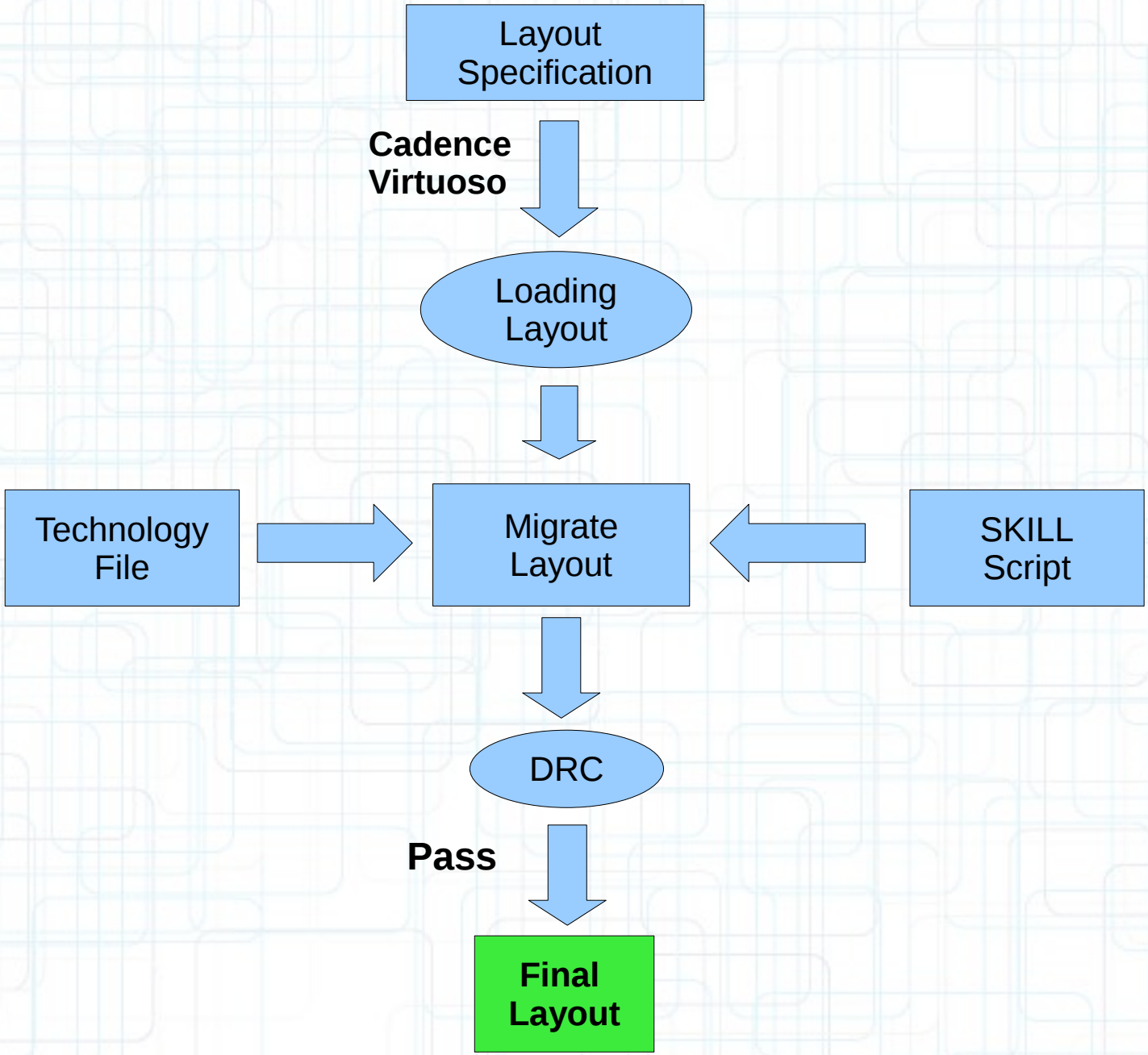
```
            # that pEnc belongs to a different class from DloGen which Group class belong to.
            # and these two classes are not communicative so it is not doable to directly assign the value

            # now create the gate contact
            gateContact=Contact(self.gateLayer, self.metalLayer,'PIN')
            # as input of inverter, use default contact width
            # the coords of gateContact is still floating

            pgroup.add(pEncPimp)   # add layer pimp into pgroup
            fgPlace(gateContact, WEST, pgroup)
            pgroupContactDist = pgroup.getBBox().left - gateContact.getBBox().right   # measure the posistion of gateContact w.r.t pgroup
            fgPlace(gateContact,EAST,ngroup)
            ngroupContactDist = gateContact.getBBox().left - ngroup.getBBox().right   # measure the position of gateContact w.r.t. ngroup
            gateContactWidth = gateContact.getBBox().right - gateContact.getBBox().left
            # till now gateContact has been put to the east of nmos

            # now place ntap
            fgPlace(ntapGroup, SOUTH, pgroup)
            pgroup.add(ntapGroup)
            pEnctmp = fgAddEnclosingRects(pgroup, [Layer('nwell')])
            pEncNwell = Rect(Layer('nwell'), pEnctmp.getBBox())     # form nwell
            pgroup.add(pEncNwell)
            pEnctmp.destroy()
            pmoveDist = ngroup.getBBox().right + ngroupContactDist + gateContactWidth + pgroupContactDist - pEncNwell.getBBox().left
            pgroup.moveTowards(EAST,pmoveDist)   # move the pmos to a proper position


            polyGap = Rect(self.gateLayer, Box(ngateRect.getBBox().right, ngateRect.getBBox().bottom, pgateRect.getBBox().left, pgateRect.get
BBox().top))   # draw poly to fill the gap between nmos and pmos gates


            npowerMetal = Rect(self.metalLayer, Box(0, 0, self.powerMetalWidth,(nEnc.getBBox().top-ptapEnc.getBBox().bottom)))     # draw nm
os power metal layer (GND), but the coods are still floating
            ppowerMetal = Rect(self.metalLayer, Box(0, 0, self.powerMetalWidth,(pEncNwell.getBBox().top-pEncNwell.getBBox().bottom)))      #
draw pmos power metal layer (VDD)

            prMetal1 = self.tech.getPhysicalRule('minSpacing', Layer('metal1'))     # get the minimum spacing of metal1
            npowerMetal.setRight(ndrainContact.getBBox().left - prMetal1.value)
            npowerMetal.setLeft(ndrainContact.getBBox().left - prMetal1.value - self.powerMetalWidth)
            npowerMetal.setTop(nEnc.getBBox().top)
            npowerMetal.setBottom(ptapContact.getBBox().bottom)
            ppowerMetal.setLeft(pdrainContact.getBBox().right + prMetal1.value)
            ppowerMetal.setRight(pdrainContact.getBBox().right + prMetal1.value + self.powerMetalWidth)
            ppowerMetal.setTop(pEncNwell.getBBox().top)
            ppowerMetal.setBottom(pEncNwell.getBBox().bottom)

            # now wire up everything
            if npowerMetal.getBox().right < nsourceContact.getBBox().left :  # npowerMetal is in the left of nsourceContact
                    npowerSourceMetal = Rect(Layer('metal1'), Box(npowerMetal.getBBox().right, nsourceContact.getBBox().bottom, nsourceContac
```

Cons

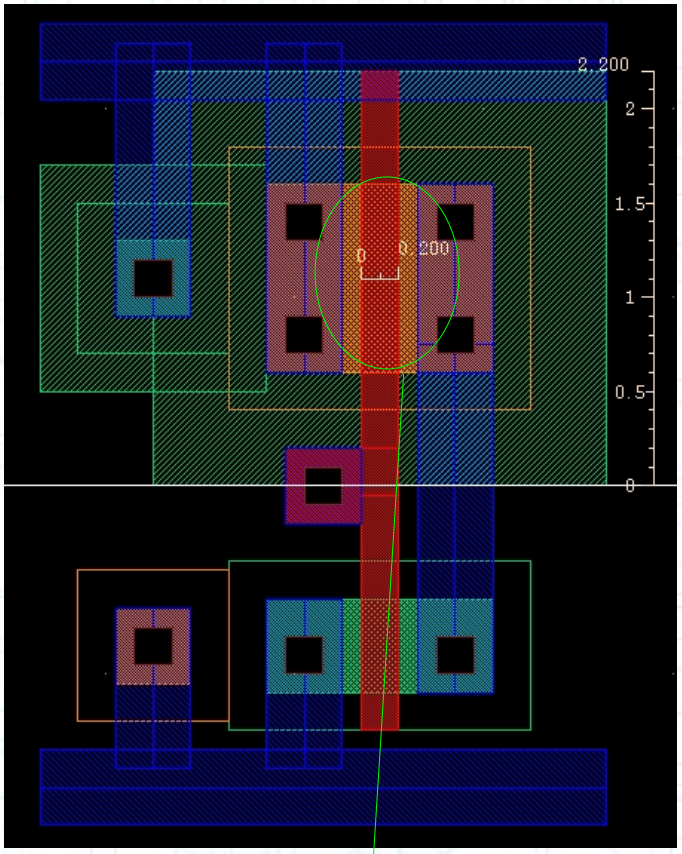
- Every rule must be accounted for
- Newly developed technology may not be accounted for
- Heavy scripting (e.g.: 300 lines for an inverter)

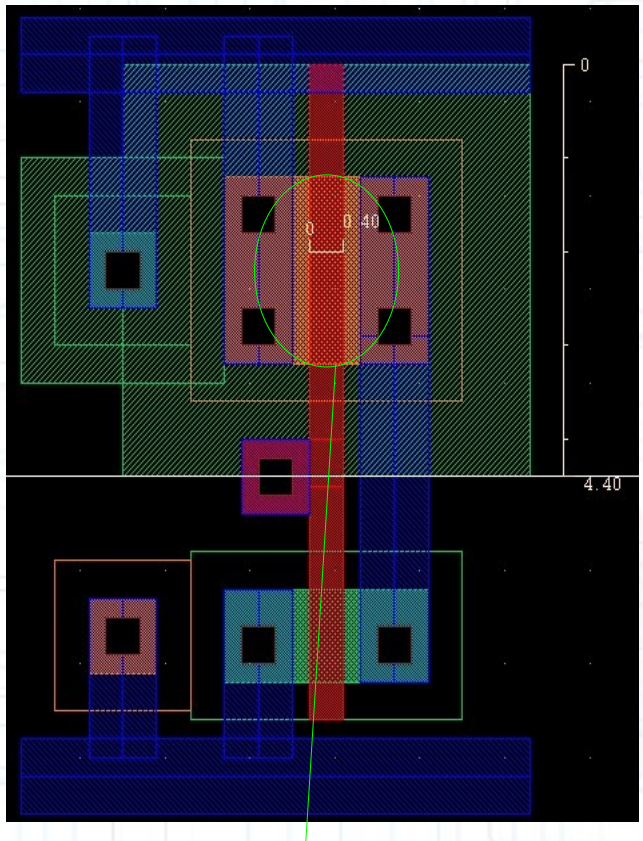
Cadence SKILL: Methodology 1

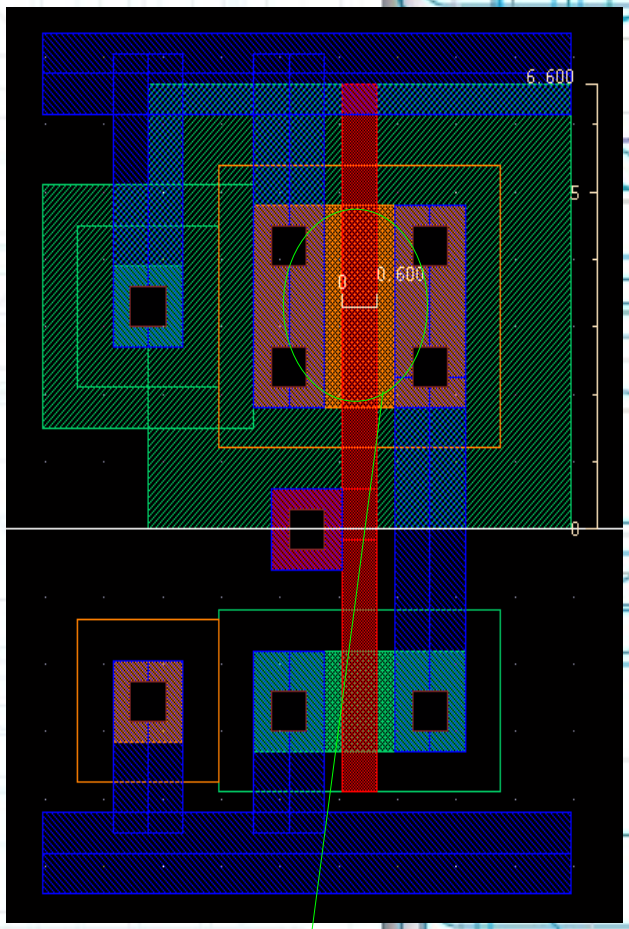
Quality By Design (QbD)

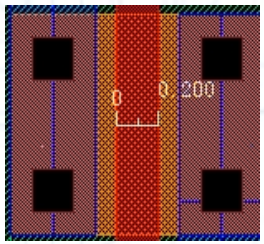


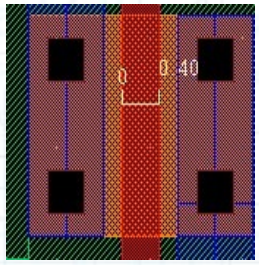
Cadence SKILL: Inverter

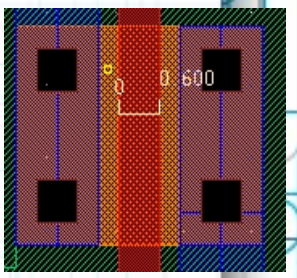




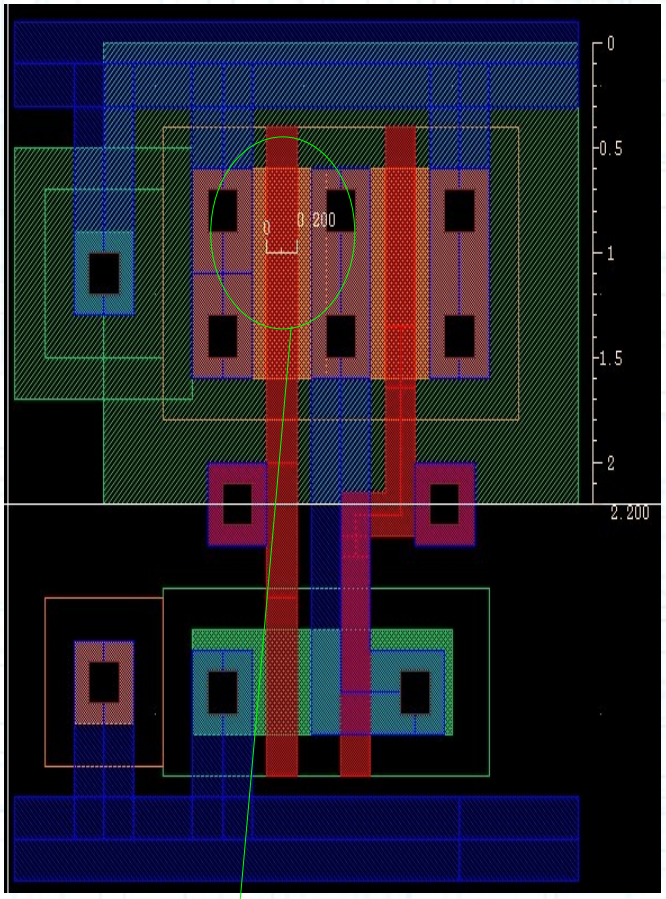


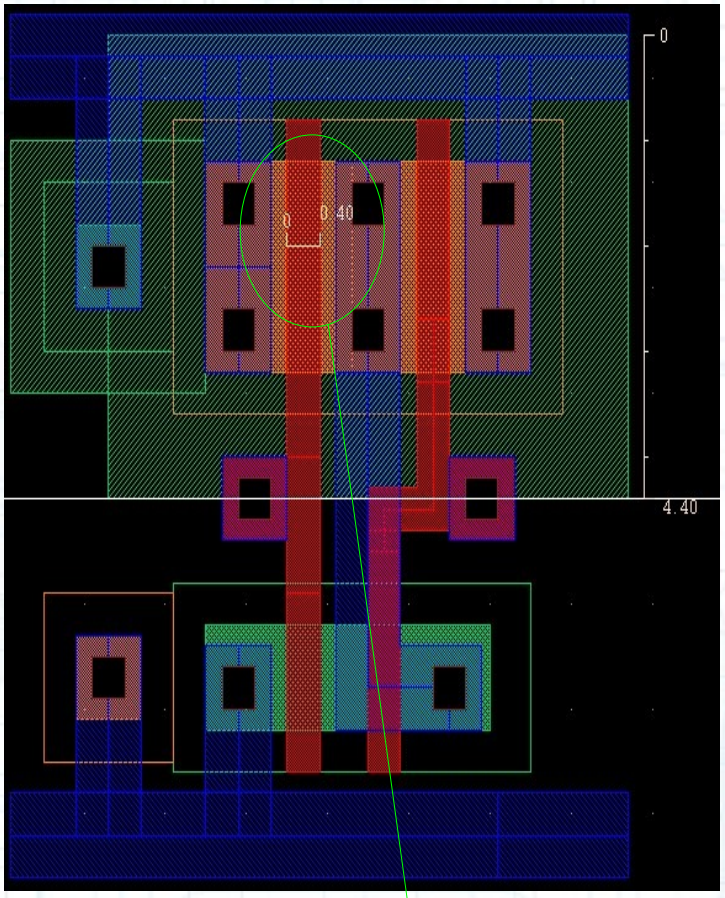


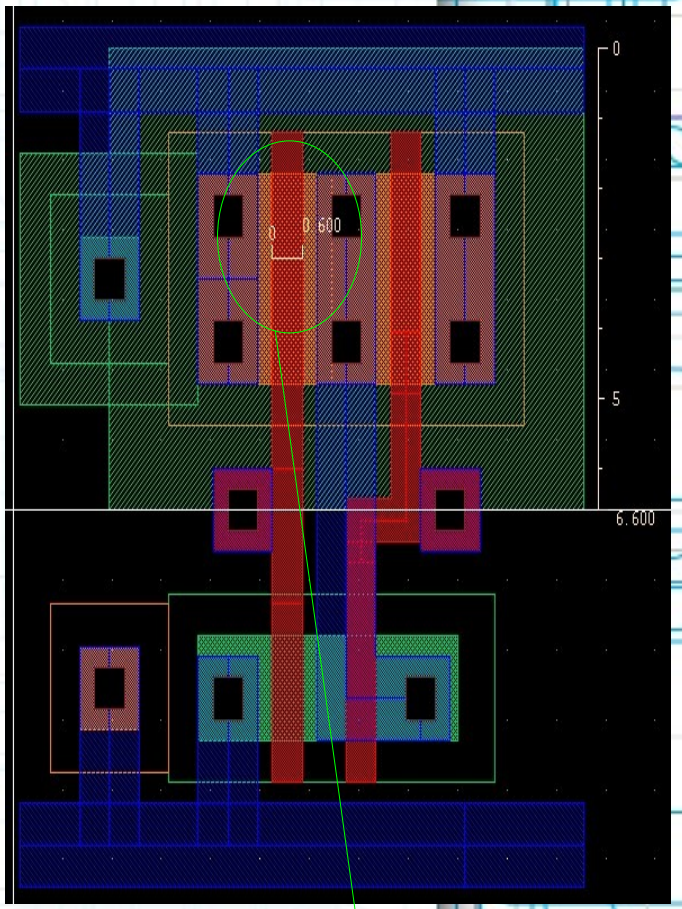


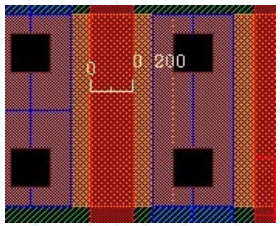


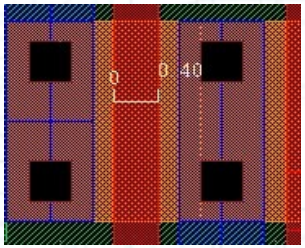
Cadence SKILL: NAND

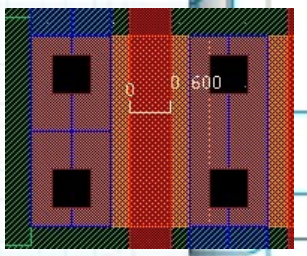




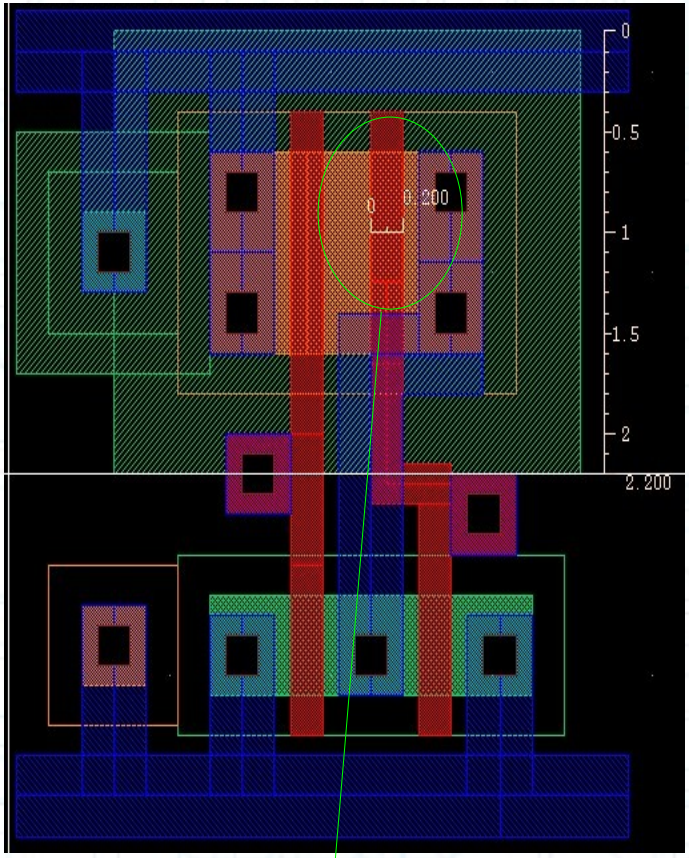


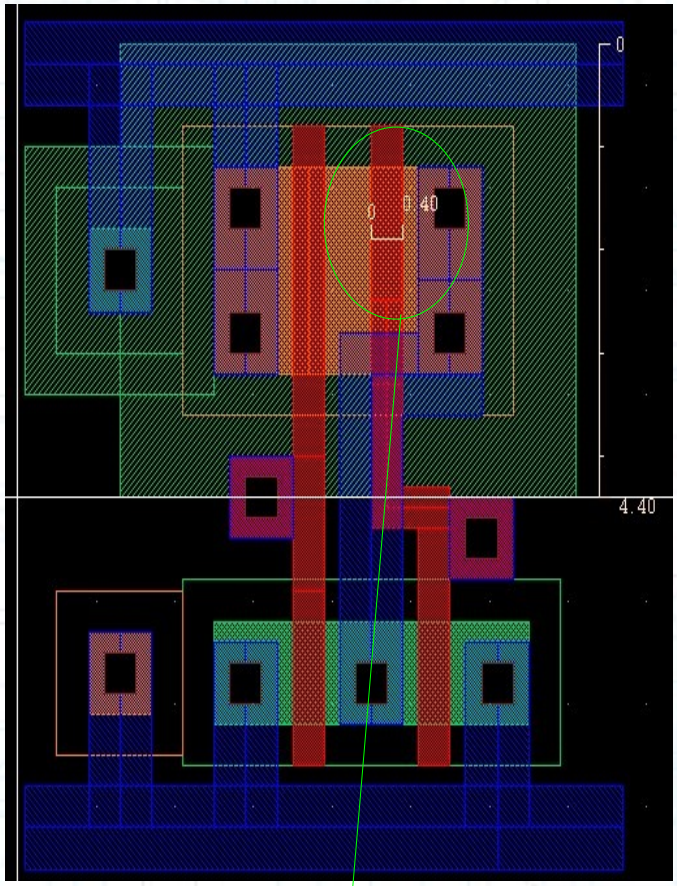


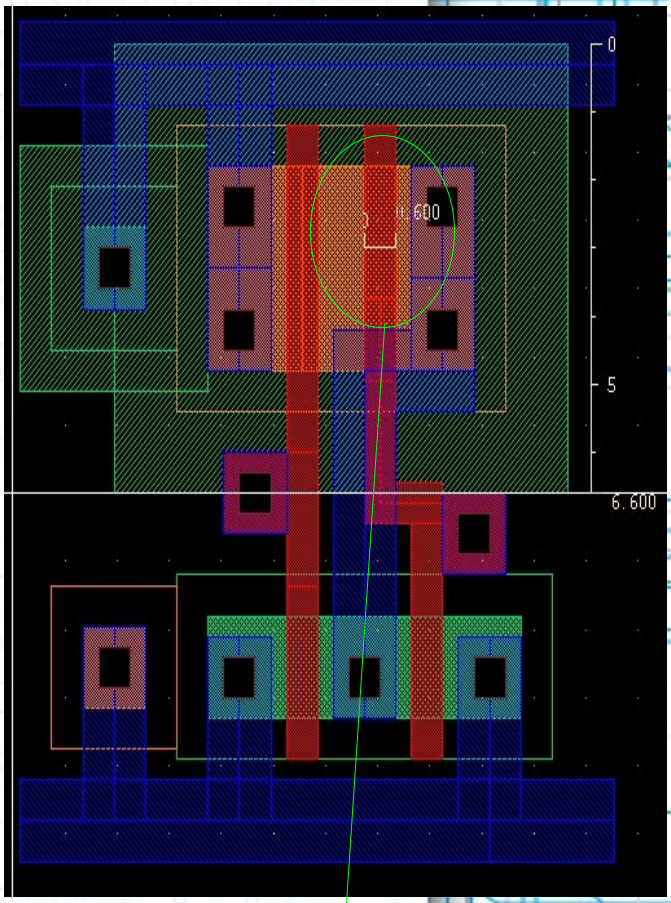


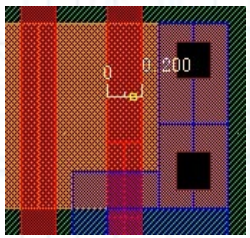


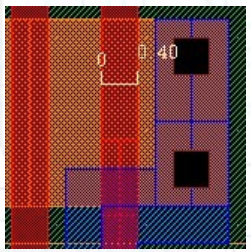
Cadence SKILL: NOR

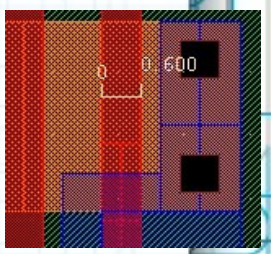








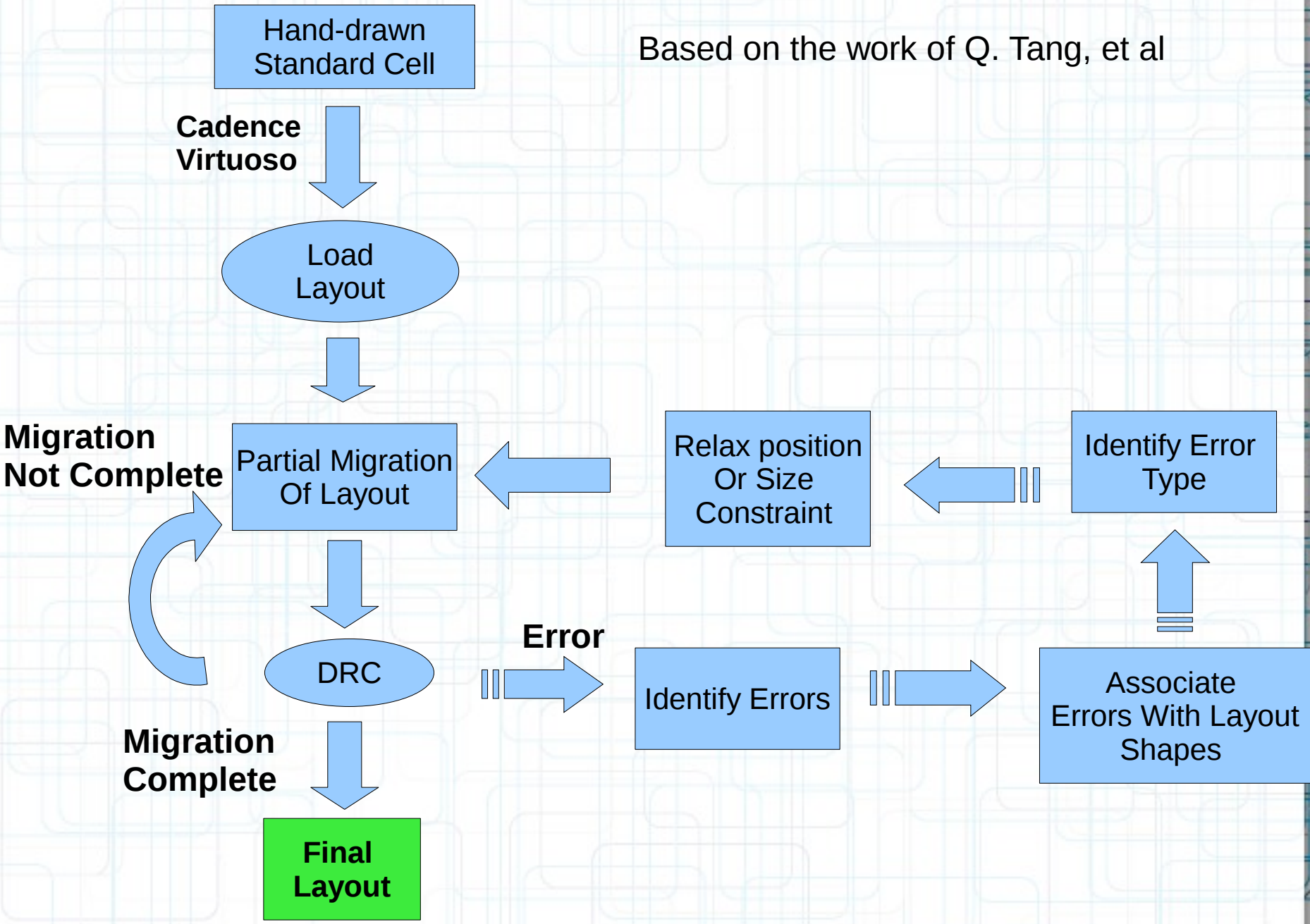




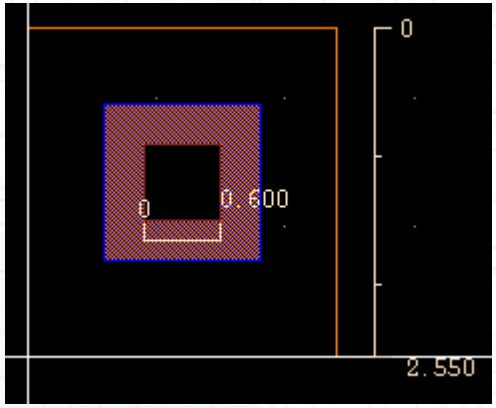
Cadence SKILL: Methodology 2

Adaptive Migration and Constraint Relaxation

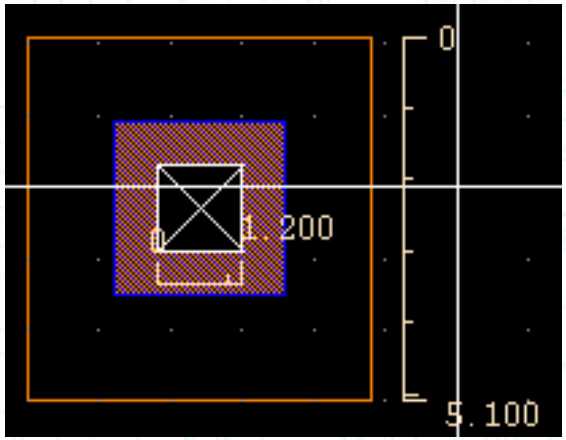
Based on the work of Q. Tang, et al



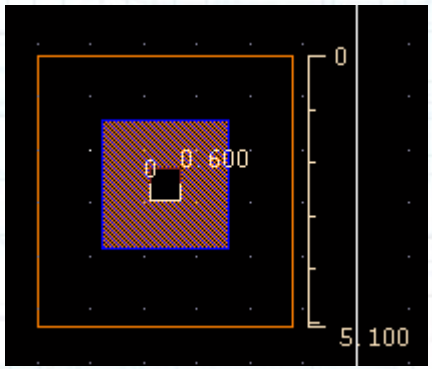
Adaptive Migration



P-Tap 0.6ami



P-Tap Simulated Technology Migration, Non-Adaptive. Error: Via must be 0.6um square.



P-Tap Simulated Technology Migration, Adaptive. Error resolved

Adaptive Migration: Pros & Cons

Pros

- Requires no additional work.
- Mature literature on the subject.
- Adapts to new design rule requirements
- Same approach may be used for optimization.
- Built in SKILL functions (Not used here).

Cons

- Adaptive is very complex.
- Large up front development.

Evaluation: Comparison of Tools

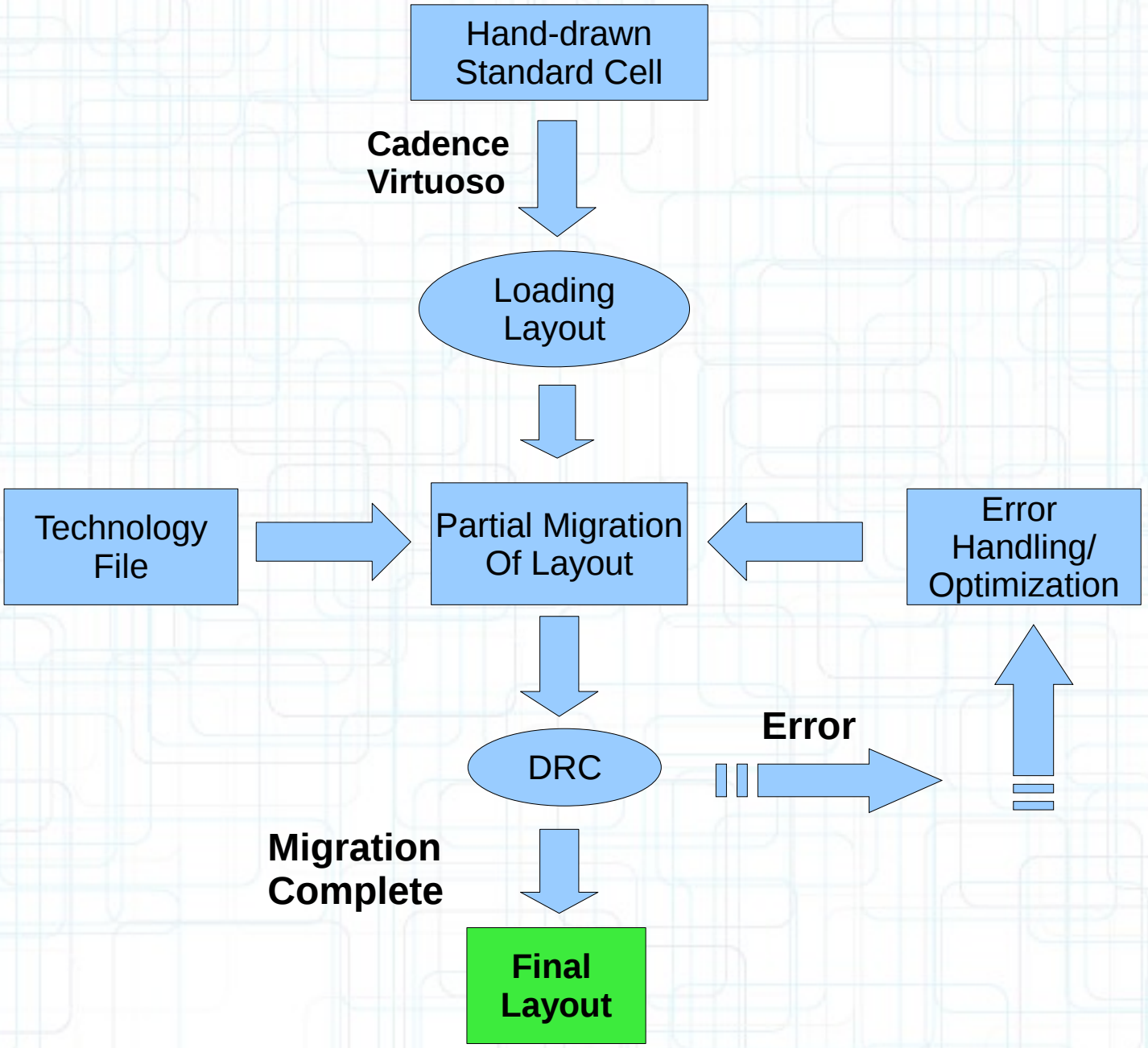
	PyCell Studio	Cadence SKILL
Scripting Language	Python	SKILL (LISP-like)
Cost	Free to use but pay to get commercial support	Pay to use and get commercial support
Extension	Almost none, have to interface with 3rd party	Good
Community support	Almost none	Good
Emphasis	Parameterized cell	Generic
Database	Open Access database	CDS Start supporting OA

* Cadence has a toolbox of doing layout compaction

Cadence is a better candidate

Evaluation: Methodology Recommendation

QbD with Adaptive Migration and Constraint Relaxation



Summary

- The motivation of this project
- Two tools: PyCell and Cadence SKILL
- Different approaches
- Evaluation

THANKS !

QUESTIONS ?